

### ***In the Claims***

The status of claims in the case is as follows:

1        1.    [Currently amended] ~~Method for~~ A method for processing  
2        a client session request received at a server in a system  
3        including a client, a server, and host with said server  
4        executing exit programs for negotiating a confirmation  
5        record on a session connection request in which direct  
6        communication between said client and said server is held on  
7        a connection for duration of a dialogue, comprising the  
8        steps of:

9                said client connecting to said server;

10               said client and said server negotiating environment  
11               parameters for establishing a connection-oriented  
12               connection of said server with said client, said client  
13               and said server communicating over said connection  
14               using a same client/server communications protocol,  
15               said client including a graphical user interface  
16               selectively assigned a session name enabling client  
17               emulator communication at an application layer with  
18               said server;

19               while negotiating said environment parameters, said  
20               server inviting said client to negotiate terminal type  
21               and submit user environment variables;

22               said client responding by returning to said server said  
23               terminal type and submitting a request for a custom  
24               confirmation record, said request including at least

25           one user variable;

26           responsive to receiving said user variable and said  
27           request for a custom confirmation record from said  
28           client, said server executing an exit program for  
29           calling and passing said user variable to a host  
30           application at said host external to said server, said  
31           host application processing said user variable and  
32           responsive thereto returning custom data to said  
33           server, said custom data selectively including a user  
34           variable received from said client that was selected  
35           and used; and

36           said server concluding negotiating said environment  
37           parameters with said client selectively including  
38           sending to said client a confirmation record ~~and custom~~  
39           ~~record~~ including said custom data received from said  
40           exit program.

2.   [Original]       The method of claim 1, said negotiating,  
inviting, and sending steps executing within the application  
layer of a TCP/IP protocol stack.

3.   [Previously presented]   The method of claim 1, further  
comprising the step responsive to a user variable requesting  
a confirmation record, sending to said client a confirmation  
record without said custom data.

4.   [Original]       The method of claim 1, said confirmation  
record including a field defining a pass through data  
length, said pass through data including said confirmation  
record and said custom data.

5. [Previously presented] The method of claim 1, further comprising the step of appending said custom data to said confirmation record.

6. [Previously presented] The method of claim 1, said request being for a default custom confirmation record, and further comprising the step of sending to said client default data received at said exit program at said server from said host application in said custom data.

7. [Previously presented] The method of claim 1, said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, further comprising the step of sending to said client defined data in said custom data.

8. [Previously presented] The method of claim 7, said sending step including executing at said server a customer defined exit program on said list to access said host to generate said defined data.

9. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data received at said exit program at said server from said host application indicia identifying a device allocated by a host server.

10. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data indicia received at said exit program at said server from said host application identifying a terminal or printer device allocated by said host.

11. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data indicia received at said exit program at said server from said host application identifying an associated device linked to a current session by a host.

12. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data indicia received at said exit program at said server from said host application identifying a physical location for receiving output.

13. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data indicia received at said exit program at said server from said host application identifying system security level and password encryption requirements.

14. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data indicia identifying another device for retrying a rejected request.

15. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data indicia identifying a reason for a failed auto-signon request.

16. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data indicia identifying a reason for denial of session connection request upon system overload and redirection to an alternate time or host.

17. [Previously presented] The method of claim 4, further comprising the step of providing in said custom data indicia received at said exit program at said server from said host application identifying custom information for interpretation by said client.

1 18. [Previously presented] A client/server system  
2 including a client, a server, and a host with said server  
3 executing exit programs for negotiating a confirmation  
4 record on a session connection request in which direct  
5 communication between said client and said server is held  
6 for duration of a dialogue, comprising:

7 a custom confirmation record;

8 a user exit program running on said server;

9 said client operating in conjunction with said user  
10 exit program for requesting said custom confirmation  
11 record from said server, and responsive thereto for  
12 engaging in subsequent client/server negotiations; said  
13 client and said server communicating over a connection-  
14 oriented connection using a same client/server  
15 communications protocol, said client including a  
16 graphical user interface selectively assigned a session  
17 name enabling client emulator communication at an  
18 application layer with said server;

19 a host application program module for receiving from  
20 said exit program a user variable provided to said  
21 server by a client request for a custom confirmation  
22 record and responsive thereto for returning to said

23 server custom data selectively including said user  
24 variable;

25 said server further for sending to said client a  
26 confirmation record including said custom data.

19. [Original] The system of claim 18, said client  
being a Telnet client.

20. [Previously presented] The system of claim 18,  
further comprising:

said client being selectively operable for negotiating  
a send-custom-confirmation-record with a 'yes', 'no' or  
defined data value; and

said user exit interpreting said data value and sending  
default or defined information received at said exit  
program at said server from said host application back  
to said client in said custom confirmation record.

21. [Previously presented] The system of claim 20, said  
custom confirmation record containing diagnostic information  
provided by said server along with custom information  
received at said exit program at said server from said host  
application by said user exit program.

22. [Previously presented] The system of claim 21, said  
custom information being provided by user exit programs  
executing in said server to call application programs at  
said host.

1     23. [Previously presented]     A method for operating a  
2     client to establish a network connection with a server in a  
3     system including a client, a server, and a host with said  
4     server executing exit programs for negotiating a  
5     confirmation record on a session connection request in which  
6     direct communication between said client and said server is  
7     held for duration of a dialogue, comprising the steps of:

8             said client connecting to said server;

9             said client negotiating with said server environment  
10            parameters for establishing a connection-oriented  
11            connection with said server, said client and said  
12            server communicating over said connection using a same  
13            client/server communications protocol, said client  
14            including a graphical user interface selectively  
15            assigned a session name enabling client emulator  
16            communication at an application layer with said server;

17            said client receiving from said server an invitation to  
18            negotiate terminal type and submit user environment  
19            variables;

20            said client responding to said invitation by requesting  
21            said server to provide a custom confirmation record,  
22            the request including at least one user variable; and

23            receiving at said client said custom confirmation  
24            record, said custom confirmation record received at  
25            said client including custom data provided by a host  
26            application program responsive to receiving said user  
27            variable from an exit program executing at said server.

24. [Previously presented] The method of claim 23, said custom confirmation record including return code, system name, device name and said custom data.

25. [Original] The method of claim 24, further comprising the steps of:

operating said server to request a custom information record from said client.

26. [Original] The method of claim 25, said request comprising an invitation to said client from said server to respond with all environment variables.

27. [Original] The method of claim 26, said client responding to said invitation by returning a custom information record as part of said environment variables.

28. [Original] The method of claim 27, said client responding to said invitation with a request that said server return to said client a custom confirmation record.

29. [Previously presented] The method of claim 28, further the steps of

operating an exit program at said server to call an application at said host to interpret the value in said custom information record to selectively return a custom confirmation record response.

30. [Previously presented] The method of claim 29,



further comprising the steps of specifying in said custom confirmation record a list of custom fields to be returned by said server.

31. [Previously presented] The method of claim 28, further comprising the steps of specifying in said custom confirmation record unstructured data for subsequent parsing and processing by said server, an application program at said host called by an exit program at said server, or an independent job.

1 32. [Currently amended] ~~Method for~~ A method for operating a  
2 client to establish a network connection with a server in a  
3 system including a client, a server, and a host with said  
4 server executing exit programs for negotiating a  
5 confirmation record on a session connection request in which  
6 direct communication between said client and said server is  
7 held for duration of a dialogue, comprising the steps of:

8       said client connecting to said server;

9       said client negotiating with said server environment  
10       parameters for establishing a connection-oriented  
11       connection with said server, said client and said  
12       server communicating over said connection using a same  
13       client/server communications protocol, said client  
14       including a graphical user interface selectively  
15       assigned a session name enabling client emulator  
16       communication at an application layer with said server;

17       while negotiating said environment parameters,  
18       receiving from said server an invitation to negotiate

19 terminal type and submit user environment variables;  
20 said client responding by returning to said server said  
21 terminal type and submitting a request for a custom  
22 confirmation record, said request including at least  
23 one user variable;  
24 responsive to sending to said server said user variable  
25 requesting a custom confirmation record, receiving at  
26 said client from said server a confirmation record and  
27 custom record data for enabling said client to engage  
28 in subsequent negotiations directly with said server,  
29 said custom record data generated by said host  
30 responsive to execution of a server exit program  
31 passing to a host application said user variable.

33. [Original] The method of claim 32, said negotiating, inviting, and sending steps executing within the application layer of a TCP/IP protocol stack.

34. [Previously presented] The method of claim 32, further comprising the step, responsive to said invitation to submit user variables, of requesting a confirmation record, and responsive thereto receiving from said server a confirmation record without said custom record data.

35. [Original] The method of claim 32, said confirmation record including a field defining a pass through data length, said pass through data including said confirmation record and said custom record data.

36. [Original] The method of claim 32, further

comprising the step of receiving said custom record data appended to said confirmation record.

37. [Original] The method of claim 32, said request being for a default custom confirmation record, and further comprising the step of receiving from said server, default data in said custom record data.

38. [Previously presented] The method of claim 32, said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, further comprising the step of receiving from said server, client defined data provided by a host application responsive a server exit program in said custom record data.

39. [Previously presented] The method of claim 38, further including the step of providing to said server a customer defined exit program accessing a host application program for parsing said list to generate said defined data.

40. [Previously presented] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a device allocated by said host application.

41. [Previously presented] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a terminal or printer device allocated by said host application.

42. [Original] The method of claim 35, further

comprising the step of receiving in said custom record data indicia identifying an associated device linked to a current session by a host.

43. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a physical location for receiving output.

44. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying system security level and password encryption requirements.

45. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying another device for retrying a rejected request.

46. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a reason for a failed auto-signon request.

47. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a reason for denial of session connection request upon system overload and redirection to an alternate time or host.

48. [Original] The method of claim 35, further comprising the step of receiving in said custom record data

indicia identifying custom information for interpretation by said client.

1     49. [Currently amended] A client system for establishing a  
2     network connection with a server in a system including a  
3     client, a server, and a host with said server executing exit  
4     programs for negotiating a confirmation record on a session  
5     connection request in which direct communication between  
6     said client and said server is held for duration of a  
7     dialogue, comprising:

8             a first logic element stored in a memory device at said  
9             client for negotiating environment parameters for  
10            establishing a connection-oriented connection with said  
11            server;

12           said parameters including a request for said server to  
13           provide a custom confirmation record to said client,  
14           said request including at least one user variable, said  
15           client including a graphical user interface selectively  
16           assigned a session name enabling client emulator  
17           communication at an application layer with said server;  
18           and

19           a second logic element stored in said memory device at  
20           said client for receiving said confirmation record from  
21           said server, said confirmation record including custom  
22           data provided to an exit program at said server by a  
23           host application external to said server for enabling  
24           said client to engage in subsequent programmable  
25           negotiations with said server, said client and said  
26           server communicating over said connection using a same

50. [Original] The system of claim 49, said custom confirmation record including return code, system name, device name and custom data.

51. [Original] The system of claim 50, further comprising:

a third logic element stored in a memory device at said server for operating said server to request a custom information record from said client.

52. [Original] The system of claim 51, said request comprising an invitation to said client from said server to respond with all environment variables.

53. [Currently amended] The system of claim 52, said client further comprising a fourth logic element stored in said memory device at said client for responding to said invitation by returning a custom information record as part of said environment variables.

54. [Currently amended] The system of claim 53, said client further comprising a fifth logic element stored in said memory device at said client for responding to said invitation with a request that said server return to said client a custom confirmation record.

55. [Currently amended] The system of claim 54, said server further comprising an exit program stored in said memory device at said server for calling an application at said

host for interpreting the value in said custom information record to selectively return a custom confirmation record response.

56. [Currently amended] The system of claim 54, further comprising a logic element stored in said memory device at said server for specifying a list of custom fields to be returned by said server in said custom confirmation record.

57. [Currently amended] The system of claim 54, further comprising a logic element stored in a memory device for specifying in said custom confirmation record unstructured data for subsequent parsing and processing by said server, an application at said host called by said exit program, or an independent job.

1 58. [Currently amended] ~~System~~ A system including a client,  
2 a server, and a host with said server executing exit  
3 programs on a session connection request for processing a  
4 client session request in which direct communication between  
5 said client and said server is held for duration of a  
6 dialogue, comprising:

7 a client memory device;

8 a host memory device;

9 a first logic element stored in said host memory device  
10 at said server for negotiating environment parameters  
11 for establishing a connection-oriented connection with  
12 said client and inviting said client to negotiate  
13 terminal type and submit user variables to said server,

14        said client including a graphical user interface  
15        selectively assigned a session name enabling client  
16        emulator communication at an application layer with  
17        said server; and  
  
18        a second logic element stored in said client memory  
19        device at said client for returning to said server said  
20        terminal type and a request for a custom confirmation  
21        record, said request including at least one user  
22        variable; and  
  
23        an exit program stored in said host memory device at  
24        said server, responsive to receiving a user variable  
25        from said client requesting a custom confirmation  
26        record, for executing an exit program stored in said  
27        host memory device requesting of an application program  
28        stored in said host memory device at said host custom  
29        data for sending to said client in a confirmation  
30        record.

59. [Original]        The system of claim 58, further  
comprising a TCP/IP protocol stack including within an  
application layer said exit program generating said custom  
record data.

60. [Currently amended] The system of claim 58, said first  
logic element further operable responsive to a user variable  
requesting a confirmation record for sending to said client  
a confirmation record without said custom record data.

61. [Original]        The system of claim 58, said  
confirmation record including a field defining a pass



through data length, said pass through data including said confirmation record and said custom record data.

62. [Currently amended] The system of claim 58, said first logic element further operable for appending said custom record data to said confirmation record.

63. [Currently amended] ~~System for~~ A system for operating a client to establish a network connection with a server in a system including a client, a server, and a host with said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, comprising:

a client memory device;

a server memory device;

a first logic element stored in said client memory device for connecting to said server and negotiating environment parameters for establishing a connection-oriented connection with said server;

a second logic element stored in said client memory device for receiving from said server an invitation to negotiate terminal type and submit user variables, said client and said server communicating over said connection using a same client/server communications protocol, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer

22           with said server;

23           a third logic element stored in said client memory  
24           device at said client for sending to said server said  
25           terminal type and submitting a request for a custom  
26           confirmation record, said request including at least  
27           one user variable; and

28           a fourth logic element stored in said client memory  
29           device for receiving from said server a confirmation  
30           record and custom record data, said custom record data  
31           generated by a host application selecting and using  
32           said user variable passed to said host by an exit  
33           program stored in said server memory device at said  
34           server.

64. [Original]       The system of claim 63, further  
comprising a TCP/IP protocol stack including an application  
layer within which said logic elements execute.

65. [Original]       The system of claim 63, further  
comprising the step responsive to said invitation to submit  
user variables, requesting a confirmation record, and  
responsive thereto receiving from said server a confirmation  
record without said custom record data.

66. [Original]       The system of claim 63, said  
confirmation record including a field defining a pass  
through data length, said pass through data including said  
confirmation record and said custom record data.

67. [Original]       The system of claim 63, said second

logic element further responsive for receiving said custom record data appended to said confirmation record.

68. [Original] The system of claim 63, said request being for a default custom confirmation record, and said second logic element further operable for receiving from said server default data in said custom record data.

69. [Original] The system of claim 63, said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, said second logic element further operable for receiving from said server client defined data in said custom record data.

70. [Previously presented] The system of claim 69, further including a logic element stored in said host memory device for providing to said server a customer defined exit program for calling an application at said host for parsing said list to generate said defined data.

1 71. [Currently amended] A physical program storage device  
2 readable by a machine, ~~tangibly embodying~~ having stored  
3 thereon a program of instructions executable by a machine to  
4 perform method steps for processing a client session request  
5 received at a server in a system including a client, a  
6 server, and a host with said server executing exit programs  
7 for negotiating a confirmation record on a session  
8 connection request in which direct communication between  
9 said client and said server is held for duration of a  
10 dialogue, said method steps comprising:

11       said client connecting to said server;

12       said client and said server negotiating environment  
13       parameters for establishing a connection-oriented  
14       connection with said client, said client and said  
15       server communicating over said connection using a same  
16       client/server communications protocol, said client  
17       including a graphical user interface selectively  
18       assigned a session name enabling client emulator  
19       communication at an application layer with said server;

20       while negotiating said environment parameters, said  
21       server inviting said client to negotiate terminal type  
22       and submit user environment variables to said server;

23       said client responding by returning to said by  
24       returning to said server said terminal ~~type~~ and type  
25       and submitting a request for a custom confirmation  
26       record, said request including at least one user  
27       variable;

28       responsive to receiving at said server said user  
29       variable and said request for a custom confirmation  
30       record, said server executing an exit program for  
31       calling and passing said user variable to a host  
32       application at said host external to said server, said  
33       host application processing said user variable and  
34       responsive thereto returning custom data to said  
35       server, said custom data selectively including a user  
36       variable received from said client that was selected  
37       and used; and

38           said server concluding negotiating said environment  
39           paramters with said client selectively including send  
40           to said client a confirmation record including said  
41           custom data received from said exit program.

72. [Original]       The program storage device of claim 71,  
said negotiating, inviting, and sending steps executing  
within the application layer of a TCP/IP protocol stack.

73. [Original]       The program storage device of claim 71,  
said method steps further comprising, responsive to a user  
variable requesting a confirmation record, sending to said  
client a confirmation record without said custom record  
data.

74. [Original]       The program storage device of claim 71,  
said confirmation record including a field defining a pass  
through data length, said pass through data including said  
confirmation record and said custom record data.

75. [Original]       The program storage device of claim 71,  
said method steps further comprising the step of appending  
said custom record data to said confirmation record.

76. [Original]       The program storage device of claim 71,  
said request being for a default custom confirmation record,  
and said method steps further comprising the step of sending  
to said client default data in said custom record data.

77. [Original]       The program storage device of claim 71,  
said request being for a defined custom confirmation record,  
said request including a list of one or more predefined

information items, and said method steps further comprising the step of sending to said client defined data in said custom record data.

78. [Original] The program storage device of claim 77, said sending step including executing a customer defined exit program on said list to generate said defined data.

79. [Previously presented] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a device allocated by a host.

80. [Previously presented] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a terminal or printer device allocated by a host.

81. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying an associated device linked to a current session by a host.

82. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a physical location for receiving output.

83. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying system security level and password encryption requirements.

84. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying another device for retrying a rejected request.

85. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a reason for a failed auto-signon request.

86. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a reason for denial of session connection request upon system overload and redirection to an alternate time or host.

87. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying custom information for interpretation by said client.

1 88. [Currently amended] A physical program storage device  
2 readable by a machine, ~~tangibly embodying~~ having stored  
3 thereon a program of instructions executable by a machine to  
4 perform method steps for operating a client to establish a  
5 network connection with a server in a system including a  
6 client, a server, and a host with said server executing exit  
7 programs for negotiating a confirmation record on a session  
8 connection request in which direct communication between  
9 said client and said server is held for duration of a  
10 dialogue, said method steps comprising:

11           said client connecting to said server;

12           said client and said server negotiating environment  
13           parameters for establishing a connection-oriented  
14           connection of said client with said server, said client  
15           including a graphical user interface selectively  
16           assigned a session name enabling client emulator  
17           communication at an application layer with said server;

18           receiving at said client from said server an invitation  
19           to negotiate terminal type and submit user environment  
20           variables, said client and said server communicating  
21           over said connection using a same client/server  
22           communications protocol;

23           said client responding by returning to said server said  
24           terminal type and submitting a request for a custom  
25           confirmation record, said request selectively including  
26           a user variable;

27           said server executing an exit program for calling and  
28           passing said user variable to a host application at  
29           said host external to said server, said host  
30           application processing said user variable and  
31           responsive thereto returning custom data to said  
32           server, said custom data selectively including a user  
33           variable received from said client that was selected  
34           and used by said host application; and

35           receiving at said client from said server a  
36           confirmation record including said custom data.



89. [Original] The program storage device of claim 88, said negotiating, inviting, and sending steps executing within the application layer of a TCP/IP protocol stack.

90. [Original] The program storage device of claim 88, said method steps further comprising the step, responsive to said invitation to submit user variables, of requesting a confirmation record, and responsive thereto receiving from said server a confirmation record without said custom record data.

91. [Original] The program storage device of claim 88, said confirmation record including a field defining a pass through data length, said pass through data including said confirmation record and said custom record data.

92. [Original] The program storage device of claim 88, said method steps further comprising the step of receiving said custom record data appended to said confirmation record.

93. [Original] The program storage device of claim 88, said request being for a default custom confirmation record, and said method steps further comprising the step of receiving from said server default data in said custom record data.

94. [Original] The program storage device of claim 88, said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, said method steps further comprising the step of receiving from said server client defined data in

said custom record data.

95. [Original]        The method of claim 94, further including the step of providing to said server a customer defined exit program for parsing said list to generate said defined data.

96. [Previously presented]    The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a device allocated by a host.

97. [Previously presented]    The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a terminal or printer device allocated by a host.

98. [Original]        The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying an associated device linked to a current session by a host.

99. [Original]        The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a physical location for receiving output.

100. [Original]       The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying system security level and password encryption requirements.

101. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying another device for retrying a rejected request.

102. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a reason for a failed auto-signon request.

103. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a reason for denial of session connection request upon system overload and redirection to an alternate time or host.

104. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying custom information for interpretation by said client.

1 105. [Currently amended] A computer program product embodied  
2 on a physical storage medium for operating a server in a  
3 network including a client, a server, and a host with said  
4 server executing exit programs for negotiating a  
5 confirmation record on a session connection request in which  
6 direct communication between said client and said server is  
7 held for duration of a dialogue, comprising:

8 ~~a physical storage medium;~~

9 first program instructions for connecting said client

10           to said server;

11           second program instructions for said client and said  
12           server to negotiate environment parameters for  
13           establishing a connection-oriented connection of said  
14           server with a client, said client including a graphical  
15           user interface selectively assigned a session name  
16           enabling client emulator communication at an  
17           application layer with said server;

18           third program instructions for said server to invite  
19           said client to negotiate terminal type and submit user  
20           environment variables to said server, said client and  
21           said server communicating over said connection using a  
22           same client/server communications protocol;

23           fourth program instructions responsive to said server  
24           receiving from said client a request for a custom  
25           confirmation record, said request including a user  
26           variable, for executing at said server an exit program  
27           for calling and passing said user variable to a host  
28           application external to said server, said host  
29           application processing said user variable and,  
30           responsive thereto, returning custom data to said  
31           server and sending to said client from said server a  
32           confirmation record including said custom data received  
33           from said exit program; and wherein

34           said first, second, third, and fourth program  
35           instructions are recorded on said physical storage  
36           medium.

1 106. [Currently amended] A computer program product  
2 embodied on a physical storage medium for operating a client  
3 in a network including a client, a server, and a host with  
4 said server executing exit programs for negotiating a  
5 confirmation record on a session connection request in which  
6 direct communication between said client and said server is  
7 held for duration of a dialogue, comprising:

8 ~~a physical program storage medium;~~

9 first program instructions for connecting said client  
10 to said server;

11 second program instructions for negotiating environment  
12 parameters for establishing a connection-oriented  
13 connection of said client with a server, said client  
14 including a graphical user interface selectively  
15 assigned a session name enabling client emulator  
16 communication at an application layer with said server;

17 third program instructions for receiving from said  
18 server at said client an invitation to negotiate  
19 terminal type and submit user variables, said client  
20 and said server communicating over said connection  
21 using a same client/server communications protocol;

22 fourth program instructions for returning to said  
23 server said terminal type and submitting a request for  
24 a custom confirmation record, said request including at  
25 least one user variable;

26 fifth program instructions responsive to said request

27           for executing an exit program at said server for  
28           calling and passing said user variable to a host  
29           application at said host external to said server, said  
30           host application processing said user variable and  
31           responsive thereto returning custom data to said  
32           server, said custom data including a user variable  
33           received from said client that was selected and used;

34           sixth program instructions for concluding negotiation  
35           of said environment parameters and for providing to  
36           said client said confirmation record and custom record  
37           data received at said said exit program from said host;  
38           and wherein

39           said first, second, third, fourth, fifth, and sixth  
40           program instructions are recorded on said physical  
41           program storage medium.